

# **PolarBug - ein effizienter Algorithmus zur reaktiven Hindernisumfahrung im Dauereinsatz**

R. D. Schraft, A. Traub, D. John, B. Graf

Fraunhofer IPA  
Nobelstr. 12  
70569 Stuttgart  
btg@ipa.fhg.de

## **1. Einleitung**

Mobile Roboter werden in naher Zukunft ein fester Bestandteil unseres täglichen Lebens sein /1/, /2/. Wenn sich mobile Roboter frei bewegen, so ist es von fundamentaler Wichtigkeit, dass sie weder mit Menschen noch mit Hindernissen kollidieren. Die Anwesenheit von Menschen darf sie zudem nicht daran hindern, ihre Aufgaben zu erfüllen. Praxistaugliche mobile Roboter müssen also in belebten und unstrukturierten Umgebungen kollisionsfrei navigieren können. Hierzu ist die Fähigkeit zur schnellen und effizienten Hindernisumfahrung unbedingt notwendig. Heutzutage stehen mit Laserscannern leistungsfähige und zuverlässige Sensoren zur 2D-Objekterkennung zur Verfügung.

In den folgenden Kapiteln wird der PolarBug-Algorithmus vorgestellt, ein neues Verfahren zur Hindernisumfahrung, basierend auf dem VisBug-Algorithmus /3/. Das Verfahren wurde speziell für die Hinderniserkennung mit einem Laserscanner, sowie für die schnelle Reaktion und Navigation in sich ständig ändernden Umgebungen konzipiert. Ein wesentlicher Unterschied zu bekannten Algorithmen ist die direkte Verarbeitung der in Polarkoordinaten vorliegenden Laserscanner-Messwerte, wodurch eine sehr hohe Effizienz des Algorithmus ermöglicht wird.

Die Zuverlässigkeit der PolarBug-Software im Dauereinsatz ist durch die Installation dreier mobiler Roboter im Museum für Kommunikation Berlin nachgewiesen. Die Roboter bewegen sich frei im Lichthof des Museums und müssen gegebenenfalls auf Besucher, die ihnen in den Weg treten, spontan reagieren.

## **2. Steuerungssoftware**

Die Steuerungssoftware der am Fraunhofer IPA entwickelten mobilen Roboter basiert auf den Softwarebibliotheken „Realtime Framework“ /8/ und „Robotics Toolbox“. Diese am Fraunhofer IPA entwickelten Softwarepakete gestatten es nicht nur, komplexe Robotersteuerungen in kurzer Zeit zu implementieren, sie erleichtern auch die Portierung der erstellten Programme auf unterschiedliche Entwicklungsplattformen bzw. Betriebssysteme. Die im Folgenden beschriebenen Methoden zur

Hinderniserkennung und -umfahrung sind Komponenten der „Robotics Toolbox“. In dieser befinden sich darüber hinaus Software-Module zur Ansteuerung und Auswertung verschiedener Typen von Navigations- und Sicherheitssensoren, zur Lokalisierung, Wegplanung, Bahninterpolation, Koordinatentransformation und Ansteuerung von Antriebsachsen.

### 3. Kollisionsvermeidung

Die unterste Schicht der kollisionsfreien Navigation ist ein Modul zur Kollisionsvermeidung. Dieses Modul bremst den Roboter ab, wenn er sich einem Hindernis nähert und stellt sicher, dass er zum Stillstand kommt, bevor er mit diesem kollidiert.

### 4. Hindernisumfahrung

Aufgabe des Hindernisumfahrungs-Moduls ist es Situationen zu vermeiden, in denen der Roboter einem Hindernis so nahe kommt, dass das Kollisionsvermeidungs-Modul aktiv wird. Um unnötiges Abbremsen und wieder anfahren des Roboters vor einem Hindernis zu vermeiden, wird durch das Modul Hindernisumfahrung vom Laserscanner detektierten Hindernissen frühzeitig ausgewichen. Es werden Daten des Laserscanners nicht nur auf dem zu fahrenden Weg des Roboters verwendet, vielmehr werden alle Messwerte des Laserscanners ausgewertet. Falls ein Hindernis im Weg des Roboters detektiert wurde, versucht der Roboter eine Zwischenposition zu finden, die ihn an dem Hindernis vorbei zu seinem Ziel bringt (Abb. 1).

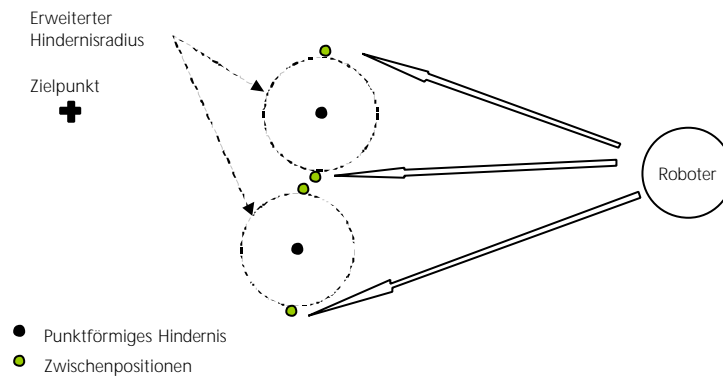


Abb. 1: Hindernisumfahrung

#### 4.1. Grundfunktionalität des PolarBug-Algorithmus

Die Vorgehensweise zur Berechnung der optimalen Zwischenposition ist in Abbildung 2 dargestellt:

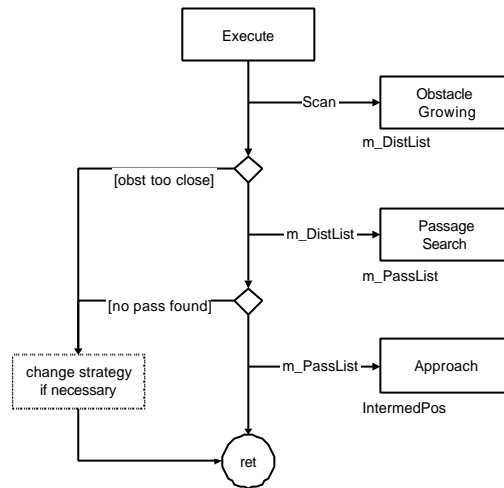


Abb. 2: Ablaufdiagramm PolarBug-Algorithmus

**ObstacleGrowing:** Ein Laserscanner sendet Laserstrahlen im Winkelabstand von  $\alpha_c$  aus. Daher liefert er als Messwerte eine diskrete, in Polarkoordinaten beschriebene Punktwolke. Beim ObstacleGrowing wird um jeden Punkt dieser Wolke ein Sicherheitskreis mit einem spezifizierten Sicherheitsradius (Radius des Roboters + Sicherheitsabstand) gezogen (Abb. 3). Des weiteren wird der Zielkreis berechnet. Dieser Kreis verläuft durch den Zielpunkt und besitzt die aktuelle Position des Roboters als Mittelpunkt. Folgende drei Werte werden für jeden Strahl des Laserscanners ermittelt:  $d_o$ : Vorderes Limit des befahrbaren Bereichs (Erster Schnittpunkt mit Kreisen um die detektierten Hindernisse),  $d_u$ : Unsicherheitszone, mögliches Ende eines Hindernisses (Hinterer Schnittpunkt mit Kreisen um die detektierten Hindernisse),  $d_e$ : Ende des Segments (Schnittpunkt mit Zielkreis oder Schnittpunkt mit Zielkreis schneidendem Sicherheitskreis)

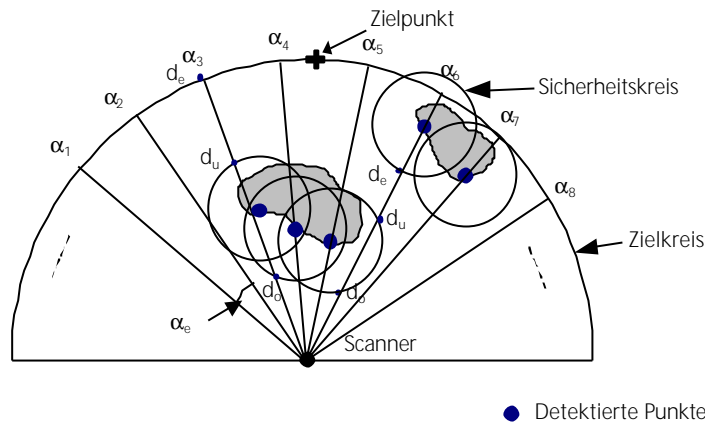


Abb. 3: ObstacleGrowing

**PassageSearch:** Ein PassageSearch ist nur notwendig, wenn die Analyse der Ausgabedaten von ObstacleGrowing ergibt, dass ein Hindernis zwischen dem Roboter und seinem Zielpunkt liegt, der Roboter seinen Zielpunkt also nicht auf direktem Wege erreichen kann. In diesem Fall werden die Ausgabedaten von ObstacleGrowing als Grundlage für die Suche nach Wegen zur Umfahrung der Hindernisse verwendet. Ausgangslinie dieser Suche ist die Verbindungslinie zwischen Roboter und Zielpunkt. Die rechts (bzw. links) dieser Linie liegenden Strahlen werden iterativ durchlaufen, bis auf jeder Seite ein Strahl gefunden wurde, bei dem  $d_o$  für den Roboter sichtbar ist. Für solche Strahlen existiert kein  $d_u$ . In Abbildung 3 sind dieses die Strahlen  $\alpha_1$  und  $\alpha_7$ . Für jede gefundene Passage werden verschiedene Werte wie Breite/Länge des unbekanntem Bereichs, Breite der Passage etc. berechnet.

**Approach:** Auf der Basis der Ergebnisse von PassageSearch wird nun unter den gefundenen Wegen der beste ausgewählt. Eine Zusammenführung der Qualitätsfaktoren für die einzelnen Passagen findet mit Hilfe von Methoden aus der Fuzzy-Logic statt. Ausgabe der Approach-Funktion ist diejenige Zwischenposition für den Roboter, welche ihn auf einem geeigneten Weg um die detektierten Hindernisse herum zu seinem Ziel führt. Bei der Bestimmung dieser Zwischenposition werden die Informationen über die Beschaffenheit der Passage verwendet, um eine möglichst optimale Zwischenposition zu ermitteln. Die Optimalitätskriterien der Zwischenposition sind zum einen ein möglichst großer Sicherheitsabstand von Hindernissen und zum andern die Minimierung des voraussichtlichen Weges zum Zielpunkt.

**ChangeStrategy:** Falls ein Hindernis so dicht vor dem Roboter ist, dass der Scanner in einem Sicherheitskreis liegt, so können keine Passagen gefunden werden. Wenn dieses der Fall ist wendet der Roboter zusätzliche Strategien an. Dies sind beispielsweise Drehen um die eigene Achse um bisher nicht sichtbare Passagen zu finden oder Ansprechen der ihn blockierenden Besucher.

## 4.2. Vorteile gegenüber herkömmlichen Methoden

Der Vorteil der vorgestellten Methode gegenüber bekannten Algorithmen wie dem VisBug /3/ oder TangentBug /6/, /7/ ist die direkte Weiterverarbeitung der vom Laserscanner in Polarkoordinaten gelieferten Daten. Des Weiteren findet bei der Planung des besten Wegs eine Berücksichtigung der Unsicherheit nichteinsehbarer Bereiche (z.B. Ausdehnung eines Hindernisses) statt. Durch den Einsatz von Fuzzy-Logik werden gefundene Passagen bewertet und die bestmögliche ausgewählt.

## 5. Zusätzliche Sicherheitssensoren

Neben dem Laserscanner verfügt jeder der Roboter, die im Museum für Kommunikation in Berlin installiert wurden, über einen Schaumstoffbumper rings um die Basis des Roboters (ca. 1 cm über Boden, ca. 50 N Auslösekraft, Sicherheitsklasse III). Wird ein Kontakt mit diesem Bumper erkannt, hält der Roboter sofort an. Um den Bereich oberhalb des Laserscanners abzusichern, werden die Signale von sechs im Bumper integrierten, nach oben gerichteten Infrarotsensoren (SHARP GP2D12, Reichweite 25 - 80 cm) ausgewertet. Zum Boden hin gerichtete Magnetsensoren werden – zusätzlich zu einer softwaregesteuerten Absicherung – verwendet, um sicherzustellen, dass keiner der Roboter den vorgeschriebenen Aktionsraum verlässt. Eine Aktivierung dieser Sensoren führt – ähnlich wie die Betätigung eines der beiden Notausknöpfe, die an jedem Roboter angebracht sind – zu einem sofortigen Stopp des Fahrzeugs.

Das beschriebene Sicherheitssystem wurde von den zuständigen Berufsgenossenschaften anerkannt. Des Weiteren konnte eine CE-Zertifizierung für die Roboter erlangt werden.

## 6. Anwendung Museumsroboter

Mit dem Ziel, Museumsbesucher mit einer neuen technischen Attraktion zu begrüßen, wurden vom Fraunhofer IPA drei neuartige mobile Roboter aufgebaut und programmiert /4/, /5/. Die mobilen Roboter bewegen sich als Teil eines künstlerischen Gesamtkonzepts frei im Lichthof des Museums für Kommunikation in Berlin. Sie unterhalten und informieren die Museumsbesucher. Einer der Roboter spielt sogar Ball (Abb. 4).

Es ist besonders wichtig, dass die Roboter bei der Erfüllung ihrer Aufgaben schnell auf die sie umgebenden Menschen reagieren, z.B. abbremsen, falls sich jemand in den Weg stellt. Besonders kritisch war die Erfüllung dieser Anforderung bei dem ballspielenden Roboter, da dieser zwar den Ball anschubsen, also mit ihm kollidieren soll, dabei aber keine Gefahr für Museumsbesucher darstellen darf. Er muss bereits reagieren, wenn ein Besucher auch nur mit dem Fuß zwischen Ball und Roboter gerät. Seit Inbetriebnahme der Museumsroboter im März 2000 konnten noch keine Kollisionen der Roboter mit einem Besucher oder der Infrastruktur des Museums beobachtet werden. Der PolarBug-Algorithmus funktioniert sicher, schnell und

zuverlässig und die Roboter versehen ihre Aufgabe tadellos. Die Roboter erreichen eine Maximalgeschwindigkeit von 0,6 m/s und haben mittlerweile in Summe über 1000 Kilometer im Museum kollisionsfrei zurückgelegt.



Abb. 4: Roboter im Einsatz

## 7. Literatur

- /1/ Wellman, P.; Krovi V.; Kumar, V.: An Adaptive Mobility System for the Disabled. Proceedings of the 1994 IEEE International Conference on Robotics and Automation, San Diego, May 8-13, 1994.
- /2/ Schraft, R.D.; Schmierer, G.: Serviceroboter – Produkte, Szenarien, Visionen. Berlin u.a., Springer Verlag, 1998
- /3/ Lumelsky, V. J.; Skewis, T.: Incorporating range sensing in the robot navigation function. IEEE Trans. Sys. Man Cybernet. 20(5), 1990, pp. 1058-1068.
- /4/ Graf, B.; Baum, W.; Traub, A.; Schraft, R. D.: Konzeption dreier Roboter zur Unterhaltung der Besucher eines Museums. In: VDI-Berichte 1552, 2000, pp. 529-536.
- /5/ Graf, B.; Schraft, R. D.; Neugebauer, J.: A Mobile Robot Platform for Assistance and Entertainment. In Proceedings of ISR-2000, Montreal, 2000.
- /6/ Rimon, E.; Kamon, I.; Canny, J. F.: Local and global Planning in Sensor Based Navigation of Mobile Robots. In Robotics Research – The Eighth International Symposium, 1998, pp. 112-123.
- /7/ Kamon, I.; Rimon, E.; Rivlin, E.: Tangent Bug: A Range-Sensor-Based Navigation Algorithm. In: International Journal of Robotic Research, 17(9), 1998, pp. 934-953.

- /8/ Traub, A.; Schraft, R. D.: An Object-Oriented Realtime Framework for Distributed Control Systems. In Proceedings of ICRA-99, 1999, pp. 3115-3121.
- /9/ Baur, R.; Feiten, W.; Lawitzky, G.: Steer angle fields: An approach to robust manoeuvring in cluttered, unknown environments. In: Robotics and Autonomous Systems , 12, 1994, pp. 209-212.
- /10/ Coombs, D.; Herman, M.; Hong, T.; Nashman, M.: Real-Time Obstacle Avoidance Using Central Flow Divergence, and Peripheral Flow. In: IEEE International Conference on Robotics and Automation, 1998, pp. 49-59.